

Mãos à obra: Fábrica de threads

Agora vamos implementar o tratamento de exceção para nossas threads.

1) Para isso, crie a classe `TratadorDeExcecao` dentro do pacote `br.com.alura.servidor` :

```
import java.lang.Thread.UncaughtExceptionHandler;

public class TratadorDeExcecao implements UncaughtExceptionHandler {

    @Override
    public void uncaughtException(Thread thread, Throwable ex) {
        System.out.println("Exceção na thread " + thread.getName() + ", " + ex.getMessage());
    }
}
```

Repare que estamos imprimindo apenas uma mensagem caso uma exceção aconteça.

2) Para poder usar a classe `TratadorDeExcecao`, crie uma nova classe `FabricaDeThreads` que representa fábrica de threads (quem diria :)). Implemente a interface `ThreadFactory` e sobrescreva o método `newThread(..)` :

```
public class FabricaDeThreads implements ThreadFactory {

    @Override
    public Thread newThread(Runnable tarefa) {

    }
}
```

3) Agora vamos implementar a fábrica. Crie um atributo estático na classe, para contar as threads criadas:

```
private static int numero = 1;
```

4) No método `newThread(..)` instancie uma nova thread, passando no construtor a tarefa (`Runnable`) e o numero de threads:

```
Thread thread = new Thread(tarefa, "Thread Servidor Tarefas " + numero);
numero++;
return thread;
```

5) Ainda no método `newThread(..)`, após o *increment*, passe para a thread o nosso `TratadorDeExcecao` e sete o thread como *daemon*

```
thread.setUncaughtExceptionHandler(new TratadorDeExcecao());
thread.setDaemon(true);
```

Segue o código completo da fábrica:

```
public class FabricaDeThreads implements ThreadFactory {  
  
    private static int numero = 1;  
  
    @Override  
    public Thread newThread(Runnable tarefa) {  
        Thread thread = new Thread(tarefa, "Thread Servidor Tarefas " + numero);  
        numero++;  
        thread.setUncaughtExceptionHandler(new TratadorDeExcecao());  
        thread.setDaemon(true);  
        return thread;  
    }  
}
```

Obs: Uma thread *daemon* é uma thread de "serviço" que será automaticamente terminada quando não houver mais nenhuma outra thread principal rodando dentro da JVM. Vimos essa funcionalidade no primeiro curso sobre threads.

6) Por último é preciso instanciar a fábrica e passar para o `threadPool`. Para isso, abra a classe `ServidorTarefas` e procure o construtor:

Substitua a linha abaixo:

```
this.threadPool = Executors.newFixedThreadPool(4);//talvez esteja usando CachedThreadPool
```

por:

```
this.threadPool = Executors.newFixedThreadPool(4, new FabricaDeThreads());
```

7) Sem nenhum console ativo no Eclipse rode a classe `ServidorTarefas` e depois o cliente. Envie comando, tudo deve continuar funcionando!